

INTELLIGENT CLUSTERING SYSTEM

By Inventors:

**Phil Delurgio
Krishna Venkatraman
Suzanne Valentine**

CROSS-REFERENCE TO RELATED APPLICATIONS

5
10 This application is a Continuation-in-Part to co-pending U.S.
Patent Application No. 10/007,002 filed November 30, 2001, entitled
“RULE RELAXATION AND SUBSET OPTIMIZATION SYSTEM”, by
Michael Neal, Krishna Venkatraman, Rob Parkin, Suzanne Valentine, Phil
Delurgio, and Hau Lee, which is incorporated by reference herein for all
15 purposes.

BACKGROUND OF THE INVENTION

The present invention relates to providing a plurality of price clusters
for a plurality of stores.

20 In businesses, prices of various products must be set. In a chain of a
plurality of stores, instead of setting the prices in each store individually, the
stores may be grouped in one or more clusters. Each cluster may form a price
zone. The creation of price zones in the past may have been done only using
geographical proximity to generate clusters.

SUMMARY OF THE INVENTION

To achieve the foregoing and other objects and in accordance with the purpose of the present invention, a method for forming a plurality of stores
5 into a plurality of clusters is provided. An optimization is performed for the plurality of stores to obtain individual store prices. The individual store prices are used to create the plurality of clusters. An optimization for the plurality of clusters is performed to obtain cluster prices.

10 In another embodiment of the invention, a method for forming a plurality of stores into a plurality of clusters is provided. Store specific information is collected. Optimized combinations for each individual store are provided based on the store specific information. Clusters are created based on the closeness of the optimal combinations.

15 In another embodiment of the invention, an apparatus for forming a plurality of stores into a plurality of clusters is provided. Computer readable code for collecting store specific information is provided. Computer readable code for optimizing combinations for each individual store based on the store specific information is provided. Computer readable code for creating clusters based on the closeness of the optimal combinations is provided.

These and other features of the present invention will be described in more detail below in the detailed description of the invention and in conjunction with the following figures.

5

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

FIG. 1 is high level flow chart of how a clustering process might be
10 used.

FIG. 2 is a high level schematic view of a price optimizing system.

FIG. 3 is more detailed flow chart of an optimization process.

FIG. 4 is a more detailed flow chart of a clustering process.

FIG. 5 is graph illustrating item prices by store.

15 FIG. 6 is a flow chart of another embodiment of the invention.

FIG.'S 7A and 7B illustrate a computer system, which forms part of a network and is suitable for implementing embodiments of the present invention.

FIG. 8 is a schematic illustration of an embodiment of the invention that functions over a network.

FIG. 9 is a flow chart of another embodiment of the invention.

FIG. 10 is a generic flow chart of an embodiment of the invention.

5

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention will now be described in detail with reference to a few preferred embodiments thereof as illustrated in the accompanying drawings. In the following description, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art, that the present invention may be practiced without some or all of these specific details. In other instances, well known process steps and/or structures have not been described in detail in order to not unnecessarily obscure the present invention.

15 In examining purchase data, it is found that geographical price zones (clustering by geography alone) may reduce profits by as much as 60%.

To facilitate understanding FIG. 10 is a broad flow chart of an embodiment of the invention. Store specific information is collected (step 1004). Such store specific information may be individual store point-of-sales data, cost data, or customer survey data. The store specific information is used

20

and processed to provide optimal combinations fore each individual store (step 1008). In the specification and claims, the use of the word "optimal" may mean a value or combination or selection that is determined through the use of an algorithm as the best result found by the algorithm. Since optimization
5 may be a complex process, the optimal result may not be the absolute best result but may be a localized best result. Clusters of stores are then formed based on the closeness of the optimal combinations for each store (step 1012). New cluster based combinations are then provided (step 1016).

To facilitate discussion, FIG. 1 is a flow chart of a process intelligently
10 clustering stores, which is an example of an embodiment of the invention. In this example store specific point-of-sales and cost data has already been collected. First, an optimization is performed using the point-of-sales and cost data(step 104). Preferably, the same optimization is performed for all stores. FIG. 2 is a schematic view of a price optimizing system 200, which may be
15 used to provide a price optimization. The price optimizing system 200 comprises an econometric engine 204, a financial model engine 208, an optimization engine 212, and a support tool 216. The econometric engine 204 is connected to the optimization engine 212, so that the output of the econometric engine 204 is an input of the optimization engine 212. The
20 financial model engine 208 is connected to the optimization engine 212, so that the output of the financial model engine 208 is an input of the optimization engine 212. The optimization engine 212 is connected to the

support tool 216 so that output of the optimization engine 212 is provided as input to the support tool 216 and output from the support tool 216 may be provided as input to the optimization engine 212. The econometric engine 204 may also exchange data with the financial model engine 208.

5 FIG. 3 is a more detailed flow chart of a preferred embodiment of a process that utilizes the price optimizing system 200 to optimize prices to perform an optimization (step 104). Data 220 is provided from the store computer systems 224 to the econometric engine 204 (step 304). Generally, the data 220 provided to the econometric engine 204 may be point-of-sale
10 information, product information, and store information. The econometric engine 204 processes the data 220 to provide demand coefficients 228 (step 308) for a set of algebraic equations that may be used to estimate demand (volume sold) given certain marketing conditions (i.e., a particular store in the chain), including a price point. The demand coefficients 228 are provided to
15 the optimization engine 212 (step 312). Additional processed data from the econometric engine 204 may also be provided to the optimization engine 212.

 The financial model engine 208 may receive data 232 from the store computer systems 224 (step 316) and processed data from the econometric engine 204. The data 232 received from the stores is generally cost-related
20 data, such as average store labor rates, average distribution center labor rates, cost of capital, the average time it takes a cashier to scan an item (or unit) of product, or how long it takes to stock a received unit of product and fixed cost

data. The financial model engine 208 may process the data to provide a variable cost and fixed cost for each unit of product in a store, generating cost coefficients (step 320). The processing by the econometric engine 204 and the processing by the financial model engine 208 may be done in parallel. Cost data 236 is provided from the financial model engine 208 to the optimization engine 212 (step 324).

The optimization engine 212 utilizes the demand coefficients 228 to create a demand equation. The optimization engine is able to forecast demand and cost for a set of prices and promotion conditions, such as store displays and ads to calculate net profit. The stores 224 may use the support tool 216 to provide optimization rules to the optimization engine 212 (step 328). The optimization engine 212 may use the demand equation, the variable and fixed costs, and the rules to compute an optimal set of prices that meets the rules (step 332). For example, if a rule specifies the maximization of profit, the optimization engine would find a set of prices that causes the largest difference between the total sales revenue and the total cost of all products being measured. In this preferred embodiment, the prices are provided on a store-by-store basis, instead of by groups of stores. In addition, in this example the prices are also provided on an item-by-item basis.

If a rule providing a promotion of one of the products by specifying a discounted price is provided, the optimization engine may provide a set of prices that allows for the promotion of the one product and the maximization

of profit under that condition. The rules normally include an objective, such as optimizing profit or optimizing volume of sales of a product and constraints, such as a limit in the variation of prices.

When profit is maximized, it may be maximized for a sum of all measured products. Such a maximization may not maximize profit for each individual product, but may instead have an ultimate objective of maximizing total profit. The result of this optimization provides an optimal product and price combination for each store.

FIG. 4 is a more detailed flow chart of a preferred embodiment of step 108 for creating clusters (groups) based on the closeness of the optimized product and price combinations for each store. Constraints may be provided (step 404). Examples of such constraints are prohibiting certain stores from being within the same cluster, forcing certain stores to be in the same cluster, setting a maximum number of clusters, and setting a minimum cluster size.

The forcing certain stores to be in the same cluster may be used to place geographically close stores in the same cluster or to place stores with similar traits, such as being in a downtown area or near a freeway exit, in the same cluster.

Stores with the closest prices and which meet the constraints are then grouped together in clusters (step 408). There may be many different approaches to clustering stores with the closest prices. Such clustering may

attempt to minimize the sum of differences or distances in prices between stores in a cluster.

FIG. 5 is a chart of an example of four stores, which are formed into two clusters, with each store carrying item X and item Y. Table 1 below
5 provides a list of the prices for items X and Y per store.

TABLE 1

STORE\ITEM	ITEM X	ITEM Y
STORE 1	\$6.00	\$10.00
STORE 2	\$6.00	\$9.00
STORE 3	\$9.00	\$7.00
STORE 4	\$8.00	\$6.00

The prices are plotted in FIG. 5 for store 1 (504), store 2 (508), store 3 (512), and store 4 (516) with the price of item X plotted along the horizontal axis and the price of item Y plotted along the vertical axis. A distance may be
10 calculated by: $Distance = \sqrt{(Price_{s,x} - Price_{s,x})^2 + (Price_{s,y} - Price_{s,y})^2}$,
where s is for one of the stores. This distance may be calculated between all possible store pair combinations. Table 2 illustrates the distances for all pairings.

TABLE 2

	DISTANCE
STORE 1 AND STORE 2	1
STORE 1 AND STORE 3	4.24
STORE 1 AND STORE 4	4.47
STORE 2 AND STORE 3	3.61
STORE 2 AND STORE 4	3.61
STORE 3 AND STORE 4	1.41

Therefore, the total distances for each possible cluster pairings are shown in Table 3.

TABLE 3

CLUSTER	TOTAL DISTANCE
1 WITH 2 AND 3 WITH 4	2.41
1 WITH 3 AND 2 WITH 4	7.85
1 WITH 4 AND 2 WITH 3	8.08

The store pair combination with the lowest distance would be chosen as the best cluster pairing. From Table 3, it can be seen that by putting stores 1 and 2 in one cluster and stores 3 and 4 in another cluster, the total distances

between stores within the same cluster are minimized, and therefore the stores are clustered accordingly.

These distances may be weighted according to volume sales by multiplying the distances by revenue.

5 Various algorithms may be used to actually form clusters. One example of such clustering is the k-means clustering method. The FASTCLUS procedure available in SAS, provided by the SAS Institute, Inc of Cary, North Carolina is one procedure that may be used to create the clusters.

10 If a store does not sell a particular product, an average price for the product may be calculated and used as the price of the product for the store that does not sell the product, so that the absence of the product does not affect the clustering of the stores. If a constraint requires that two or more stores be placed in the same cluster, their prices may be averaged together and the average price may be used as individual store prices for these stores, so that
15 the clustering algorithm automatically places these stores in the same cluster, since their prices are identical.

 After the clusters are created (step 108), prices for items in the store are set by cluster so that stores within the same clusters have items with the same prices (step 112). Setting the product and price combinations by cluster
20 is an example of providing cluster based combinations (step 1016). Preferably, the new prices may be obtained by performing a new optimization, where prices are calculated by cluster instead of by store. Such an

optimization might be weighted by volume of sales for each store. Table 4 is an example of such an optimization by cluster.

TABLE 4

STORE\ITEM	ITEM X	ITEM Y
STORE 1	\$6.00	\$9.60
STORE 2	\$6.00	\$9.60
STORE 3	\$8.50	\$6.50
STORE 4	\$8.50	\$6.50

5 The prices for the items in store 1 are the same as the prices for store 2, since they are in the same cluster. Similarly, the prices for the items in store 3 are the same as the prices for store 4, since they are in the same cluster. In this example, the prices for the cluster are close to and in between the prices for the individual stores in the cluster derived when the optimization was done on
10 a store-by-store basis. By providing clusters so the prices for each cluster are as close as possible to the prices for an optimization done on a store-by-store basis, the optimization by cluster is kept as close to the optimization on a store-by-store basis, which minimizes the loss of optimization opportunity.

Over time, as more sales data is collected, the new data may be added to iteratively change the clusters over time. Such iteration may limit the number of stores that can change clusters or may have a similar constraint.

In an embodiment of the invention, each store may have a plurality of product categories. Clustering may be according to product categories, so that a store may be in one cluster for one product category and in another cluster for another product category. This clustering may allow more constraints. For example, a product category with perishable items may have a constraint so that stores within a cluster are relatively close together, while a product category with non-perishable items may not have such a constraint.

In another embodiment of the invention, it may be found that for some product categories there may be more products available than what may be offered in each store. For example, there may be more than 500 different soap products, although the stores may only have room to sale 200 products. An optimization engine as described above may be used to determine the most profitable combination of products for each store. A chain of 100 stores may have a hundred product combinations. It may be desirable to provide five clusters of product combinations. This embodiment of the invention would provide clusters of stores according to product combinations (assortment zones) for an entire store or a product category. FIG. 6 is a flow chart on a broader embodiment of the invention. Store specific point-of-sales or survey data may be collected. Using the above example of soap products, an optimization may be performed using the store specific data to determine a

combination of 200 products of the 500 soap products that would optimize sales on a store by store bases for each of the 100 stores (step 604). The stores are then clustered into five clusters so that each cluster that a store is placed has the most similar combination of products (step 608). Stores are then

5 provided an assortment of products (product combinations) according to the cluster that the store is in (step 612). As mentioned before, such clustering would have lower sales and/or profit than providing individual product combinations for each individual store, but would be designed to maximize sales and/or revenue for different combinations of five clusters. In one

10 embodiment, the prices are set or optimized on a store by store basis. In another embodiment, an optimization may be done where stores in the same assortment zone form price clusters. Stores may be placed in an assortment cluster so that all stores in the same assortment cluster receive the same assortment and also in a price cluster so that all stores in the same price cluster

15 have the same prices.

FIG. 9 is a flow chart of another embodiment of the invention. In this example store specific information may be collected by a survey (step 954). The survey data is used to provide a store by store promotion combination optimization (step 958). In one example, 200 products may discussed in the

20 survey. A promotion combination may be a combination of 20 of the 200 products at different discounts and displays. It may be found that each store of 100 stores in a chain has a different promotion combination optimization. Promotion clusters are created to cluster the stores according to how close

their promotion combinations are (step 962). Promotion combinations for each store are then determined by the cluster that the store is in (step 966)

Other embodiments may be generated according to margins or by metric or according to direction of movement.

5 FIG.'S 7A and 7B illustrate a computer system 900, which is suitable for implementing embodiments of the present invention. FIG. 7A shows one possible physical form of the computer system. Of course, the computer system may have many physical forms ranging from an integrated circuit, a printed circuit board, and a small handheld device up to a huge super
10 computer. Computer system 900 includes a monitor 902, a display 904, a housing 906, a disk drive 908, a keyboard 910, and a mouse 912. Disk 914 is a computer-readable medium used to transfer data to and from computer system 900.

FIG. 7B is an example of a block diagram for computer system 900.
15 Attached to system bus 920 are a wide variety of subsystems. Processor(s) 922 (also referred to as central processing units, or CPUs) are coupled to storage devices, including memory 924. Memory 924 includes random access memory (RAM) and read-only memory (ROM). As is well known in the art, ROM acts to transfer data and instructions uni-directionally to the CPU and
20 RAM is used typically to transfer data and instructions in a bi-directional manner. Both of these types of memories may include any suitable type of the computer-readable media described below. A fixed disk 926 is also coupled

bi-directionally to CPU 922; it provides additional data storage capacity and may also include any of the computer-readable media described below. Fixed disk 926 may be used to store programs, data, and the like and is typically a secondary storage medium (such as a hard disk) that is slower than primary storage. It will be appreciated that the information retained within fixed disk 926 may, in appropriate cases, be incorporated in standard fashion as virtual memory in memory 924. Removable disk 914 may take the form of any of the computer-readable media described below.

CPU 922 is also coupled to a variety of input/output devices, such as display 904, keyboard 910, mouse 912 and speakers 930. In general, an input/output device may be any of: video displays, track balls, mice, keyboards, microphones, touch-sensitive displays, transducer card readers, magnetic or paper tape readers, tablets, styluses, voice or handwriting recognizers, biometrics readers, or other computers. CPU 922 optionally may be coupled to another computer or telecommunications network using network interface 940. With such a network interface, it is contemplated that the CPU might receive information from the network, or might output information to the network in the course of performing the above-described method steps. Furthermore, method embodiments of the present invention may execute solely upon CPU 922 or may execute over a network such as the Internet in conjunction with a remote CPU that shares a portion of the processing.

In addition, embodiments of the present invention further relate to computer storage products with a computer-readable medium that have computer code thereon for performing various computer-implemented operations. The media and computer code may be those specially designed and constructed for the purposes of the present invention, or they may be of the kind well known and available to those having skill in the computer software arts. Examples of computer-readable media include, but are not limited to: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROMs and holographic devices; magneto-optical media such as floptical disks; and hardware devices that are specially configured to store and execute program code, such as application-specific integrated circuits (ASICs), programmable logic devices (PLDs) and ROM and RAM devices. Examples of computer code include machine code, such as produced by a compiler, and files containing higher level code that are executed by a computer using an interpreter. Computer readable media may also be computer code transmitted by a computer data signal embodied in a carrier wave and representing a sequence of instructions that are executable by a processor.

FIG. 8 is a schematic illustration of an embodiment of the invention that functions over a computer network 800. The network 800 may be a local area network (LAN) or a wide area network (WAN). An example of a LAN is a private network used by a mid-sized company with a building complex.

Publicly accessible WANs include the Internet, cellular telephone network, satellite systems and plain-old-telephone systems (POTS). Examples of private WANs include those used by multi-national corporations for their internal information system needs. The network 800 may also be a
5 combination of private and/or public LANs and/or WANs. In such an embodiment, the price optimizing system 200 is connected to the network 800. Computer systems used by the stores 224 are also connected to the network 800. The computer systems for the stores 224 are able to bi-directionally communicate with the price optimizing system 200 over the network 800.

10 Computer readable media may be used by the computer system 900 to communicate over the network 800 to collect store specific information to generate a consumer response model, and use the consumer response model to create clusters. Computer readable media may be used by the computer system 900 to perform an optimization to obtain individual store prices, which
15 are used to create clusters and then to perform an optimization for the plurality of clusters to obtain cluster prices.

While this invention has been described in terms of several preferred embodiments, there are alterations, modifications, permutations, and substitute equivalents, which fall within the scope of this invention. It should also be
20 noted that there are many alternative ways of implementing the methods and apparatuses of the present invention. It is therefore intended that the following appended claims be interpreted as including all such alterations, modifications,

permutations, and substitute equivalents as fall within the true spirit and scope of the present invention.

11/11/11 11:11:11